

REMARKS/ARGUMENTS

Claims 40-58 are pending in the present application. Claims 21-39 are cancelled and replaced with new claims 40-58. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 102, Anticipation

The examiner has rejected claims 21-39 under 35 U.S.C. § 102 as being anticipated by *Kelkar et al.*, U.S. Patent 7,058,846, (June 6, 2006) (hereinafter, "*Kelkar*"). This rejection is respectfully traversed. With regard to claim 21, the examiner states:

In regard to claim 21 *Kelkar* discloses a method for replicating a software application in a multi-computer architecture (cluster), whereas said software application may be executed before hand on a first computer of said cluster forming a primary node and intended for replication on at least one other computer of said cluster forming a secondary node, comprising a replication of the resources associated with said software application, characterised in that the replicated resources include: (Figure 2) and (abstract) [This, figure shows two nodes 110A and 110B]

- the virtual memory of each process affected as well as its calling stack,
- system resources (inter-process communication, network connection, etc.) and
- data written on disks. (Summary; " These operations include storage management services that allow configuration changes to be made dynamically to storage resources") [Examiner reminds Applicant that resources include virtual memory, stack calls, system resources, and data written on disk.] and in that it includes on the flow updating of the replicated resources by a dynamic introspection mechanism supplying the structure of the application to be replicated, as well as a dynamic graph of the resources and dependencies implemented. (Column 3; lines 33-36) [real time is on the flow]

Office Action dated October 2, 2007.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). In this case, each and every feature of the presently claimed invention is not identically shown in the cited reference, arranged as they are in the claims.

Claim 21 (renumbered 40) is stated as follows:

A computer implemented method for replicating a software application in a multi-computer architecture cluster, the computer implemented method comprising:

executing the software application on a first computer of the cluster forming a primary node;

replicating the resources associated with the software application on at least one other computer of the cluster forming a secondary node, the resources associated with the software application on the secondary node comprising a virtual memory and calling stack of each process affected, system resources comprising inter-process communication, network connection, and data written on disks; and

updating the replicated resources incrementally, using a dynamic introspection mechanism supplying the structure of the application to be replicated and also supplying a dynamic graph of the resources and dependencies implemented.

Under the standards of *In re Bond*, *Kelkar* does not anticipate claim 40 because *Kelkar* does not teach the claimed features of “replicating the resources associated with the software application on at least one other computer of the cluster forming a secondary node, the resources associated with the software application on the secondary node comprising a virtual memory and calling stack of each process affected, system resources comprising inter-process communication, network connection, and data written on disks” and “updating the replicated resources incrementally, using a dynamic introspection mechanism supplying the structure of the application to be replicated and also supplying a dynamic graph of the resources and dependencies implemented.”

I.A. The Reference Does Not Teach or Suggest All of the Features of Claim 21

Regarding the claimed feature of “replicating the resources associated with the software application on at least one other computer of the cluster forming a secondary node, the resources associated with the software application on the secondary node comprising a virtual memory and calling stack of each process affected, system resources comprising inter-process communication, network connection, and data written on disks,” based on the rejection of claim 21, the examiner appears to believe that Figure 2 and the following portion of *Kelkar*, teaches the claimed feature:

A method, system, and computer program product to enable other nodes in a cluster to resume operations of a failed node. These operations include storage management services that allow configuration changes to be made dynamically to storage resources. Resource configuration data are synchronized on a set of nodes in a cluster immediately when a resource configuration change is made. If a node that has made a resource configuration change fails, the resource configuration change is available for use by other nodes in the set, each of which can resume operations of the failed node.

Kelkar abstract.

Kelkar is directed toward cluster failover for storage management services. *Kelkar* teaches in the abstract, “If a node that has made a resource configuration change fails, the resource configuration change is available for use by other nodes in the set, each of which can resume operations of the failed node.” *Kelkar* therefore deals with storage management resources. Figure 2 shows two nodes, reference numerals 110A and 110B sharing a storage resource, reference numeral 140. The configuration information *only* is replicated in *Kelkar*.

In contrast, the claimed invention provides replication of the resources associated with the application. *Kelkar* provides the definitions and not the resources. *Kelkar* teaches at col. 5 lines 62-64, “Resource agents 314A and 314B are present on both nodes 110A and 110B, but only one of resource agents 314A and 314B actively manages a given resource online. Other resource agents for that given resource consider the resource to be offline on their respective nodes,” showing that the resources are not available to both nodes concurrently. When a node fails, *Kelkar* teaches processing of the configuration information to activate a resource as disclosed:

In response to the failover, in action 6.3, cluster manager 330B activates resources. In one embodiment, cluster manager 330B calls an "online" entry point, or set of instructions, for each agent. Also in action 6.3, storage resource agent 416A, log agent 418B, and recovery agent 419B begin preparation to actively manage resources, such as storage resource 140. In action 6.4, each of storage resource agent 416A, log agent 418B, and recovery agent 419B requests resource configuration manager 360B for respective attributes. Resource configuration manager 360B obtains the attributes from storage resource attributes 476B, log attributes 478B, and recovery attributes 479B of resource configuration data 370B in action 6.5. Because resource configuration data 370A and 370B are synchronized, cluster manager 330B has access to current values for the attributes for storage resource 140.

In action 6.6, resource configuration manager 360B provides storage resource attributes 476B, log attributes 478B, and recovery attributes 479B to the respective agents storage resource agent 416A, log agent 418B, and recovery agent 419B. In action 6.7, storage resource agent 416B uses storage resource attributes 476B to create storage group definition 140D on node 110B.

Kelkar col. 8 lines 45-67, (emphasis provided).

Therefore, *Kelkar* teaches making the resource *definitions* available for use on another node. In contrast, the claimed feature makes the *actual resources* available on the other node as a replica. These two features are not the same. Therefore, *Kelkar* does not teach or suggest the claimed feature and therefore fails to anticipate claim 40. Accordingly, under the standards of *In re Bond*, *Kelkar* does not anticipate claim 40 or any other claim.

I.A.i. Regarding the feature of “replicating the resources associated with the software application on at least one other computer of the cluster forming a secondary node, the resources associated with the software application on the secondary node comprising a virtual memory and calling stack of each process affected, system resources comprising inter-process communication, network connection, and data written on disks,” based on the rejection of claim 21, the examiner appears to believe that the following portion of *Kelkar* teaches the claimed feature:

These operations include storage management services that allow configuration changes to be made dynamically to storage resources.

Kelkar summary.

The examiner further states “[Examiner reminds Appellant that resources include virtual memory, stack calls, system resources, and data written on disk.]” The teaching of *Kelkar* as shown previously does not make the resources available, only the definitions, which later when activated cause the resource to be created. *Kelkar* “allows configuration changes to be made” subsequent to a request after a node failure. Definitions of resources being shared as disclosed in the teaching of *Kelkar* is not to be considered the same as replicated resources in the claimed feature. Therefore, *Kelkar* does not provide the claimed feature and therefore does not anticipate claim 40.

I.A.ii With regard to the claimed feature of “updating the replicated resources incrementally, using a dynamic introspection mechanism supplying the structure of the application to be replicated and also supplying a dynamic graph of the resources and dependencies implemented,” the examiner believes the following portion of *Kelkar* provides the claimed feature:

The present invention provides a method, system, and computer program product to make resource configuration information available to nodes in a cluster in as close to real-time as possible with minimal overhead.

Kelkar col. 3 lines 33-36.

Kelkar teaches making configuration information available to nodes as soon as possible. The only similarity between the claimed feature and *Kelkar* is the use of the term “real time.” The cited reference clearly states *Kelkar* makes available “resource configuration information.” In contrast, the claimed feature provides “*updating of the replicated resources* by a dynamic introspection mechanism supplying the structure of the application to be replicated, and also supplying a dynamic graph of the resources and dependencies implemented.” Again, as shown previously *Kelkar*’s focus is on definitions and not actual resources, therefore *Kelkar* does not

provide the claimed feature. Therefore, *Kelkar* does not anticipate claim 40. Accordingly, under the standards of *In re Bond*, *Kelkar* does not anticipate claim 40.

Since independent claim 58 claims similar features, claim 58 is also distinguished from the teaching of *Kelkar* and overcomes the examiner's rejection. Since claims 41-57 depend from claim 40, the same distinctions between *Kelkar* and the claimed invention in claim 40 apply equally well for these claims. Accordingly, the rejection of previous claims 21-39 under 35 U.S.C. § 102 has been overcome.

II. Conclusion

The subject application is patentable over the cited references and should now be in condition for allowance. The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: January 2, 2008

Respectfully submitted,

/Theodore D. Fay III/

Theodore D. Fay III
Reg. No. 48,504
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants